



# APPLICATION MOBILE

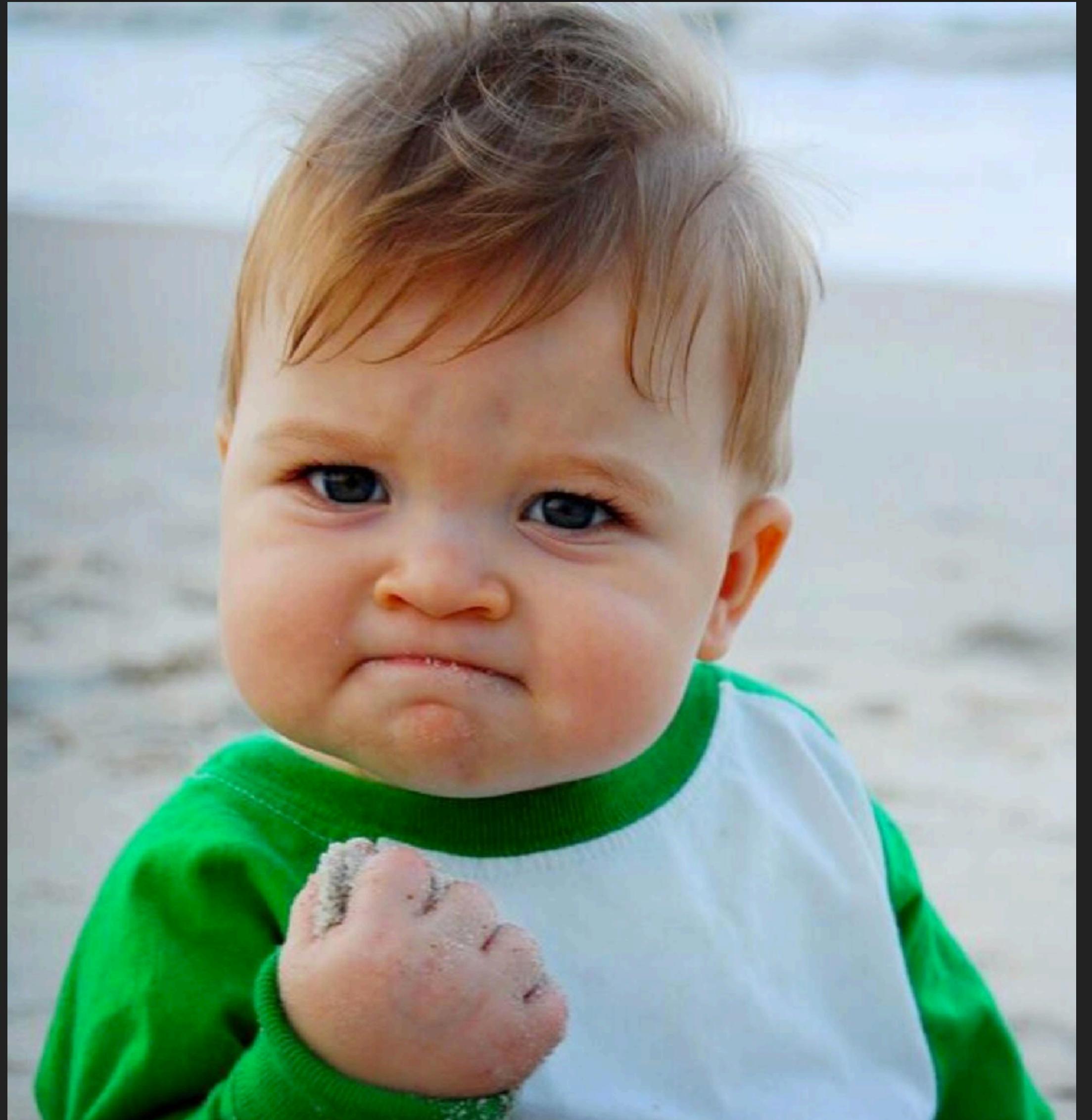
Julien Catania - [jcatania.io](http://jcatania.io)

@jcataniadotio

**y**nov  
CAMPUS

## OBJECTIFS PERSONNELS

- ▶ Aucune limite !
- ▶ Cerveau en effervescence
- ▶ Être à l'affut du moindre besoin



# QUI SUIS-JE ?





SCAN ME

QUI ÊTES-VOUS ?

# LES BASES DU WEB

## LES FICHIERS DU WEB

- ▶ HTML: HyperText Markup Language
  - ▶ Représentation de la page web
- ▶ CSS: Cascading Style Sheets
  - ▶ Décrit la présentation HTML
- ▶ JS: JavaScript
  - ▶ Intelligence de la page web

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
    <title>CSS3</title>
    <link href="style.css" rel="stylesheet" type="text/css">
  </head>
  <body>
    <div id="container"> <!-- Main container -->
      <div class="menu"> <!-- Menu -->
        <ul>
          <li><a href="#">CSS3 & HTML5 are&lt;u>cool!</u></a></li>
        </ul>
      </div>
      <div class="article"> <!-- Article -->
        <h2>CSS3 & HTML5 are cool! </h2>
        <p>This is a sample article. It contains some text and images. The text is styled using CSS3 properties like font-size, font-weight, color, and background-color. The images are styled using CSS3 properties like width, height, border, and opacity. The overall layout is defined by the CSS3 properties of the container and menu classes.</p>
        
      </div>
    </div>
  </body>
</html>
```

# CONVENTION D'ÉCRITURE

- ▶ PascalCase/UppercamelCase
  - ▶ Classe
- ▶ camelCase/lowerCamelCase
  - ▶ Attribut
  - ▶ Paramètre
  - ▶ Variable
- ▶ kebab-case/spinal-case
  - ▶ Class et ID HTML et CSS
  - ▶ Base de données
  - ▶ URL
- ▶ snake\_case/underscore\_case
  - ▶ Variable en Python, PHP, Ruby
- ▶ SCREAMING\_SNAKE\_CASE
  - ▶ Constante

# ORDRE DE PRIORITÉ DES SÉLECTEURS CSS

- ▶ Priorités des sélecteurs simples
  - ▶ Tag
  - ▶ Class
  - ▶ ID
- ▶ Priorités des sélecteurs complexes
  - (Plus il y a de sélecteur plus il sera prioritaire)
  - ▶ div.titre
  - ▶ div.titre.nom
- ▶ Dernier recours:  

```
.titre{  
background-color: red; !important;  
}
```

JAMAIS

# APPROCHES CSS

# CRÉATION DE CLASSE PARTICULIÈRES

```
.titre{  
    text-align: center;  
    color: #000;  
    background-color: #eee;  
}  
  
<h1 class="titre">Hello Ynov</h1>
```

# CRÉATION DE CLASSE GÉNÉRIQUES

```
.text-align-center{
```

```
    text-align: center;
```

```
}
```

```
.color-black{
```

```
    color:#000;
```

```
}
```

```
.background-eee{
```

```
    background-color: #eee;
```

```
}
```

```
<h1 class="text-align-center color-black  
background-eee">Hello yNov</h1>
```

# COMPORTEMENTS DE BASE CSS

- ▶ Position
  - ▶ Relative
  - ▶ Absolue
  - ▶ Fixed
  - ▶ Sticky
- ▶ Affichage (display)
  - ▶ Block
  - ▶ Inline
  - ▶ None
  - ▶ inline-block
  - ▶ Table
  - ▶ Flex

## @MEDIA

- ▶ embossed
- ▶ screen
  - ▶ Imprimantes braille
  - ▶ Écrans
- ▶ projection
- ▶ handheld
- ▶ projectors (objets de présentation avec slides)
- ▶ print
  - ▶ Impression
  - ▶ Terminal/police à pas fixe
- ▶ aural (CSS 2.0) / speech (CSS 2.1)
- ▶ tv
  - ▶ Synthèses vocales
- ▶ braille
  - ▶ Téléviseur
  - ▶ Plages braille
- ▶ all
- ▶ Tous les précédents

Ecran entre 200 et 640 pixels :

```
@media screen and (min-width: 200px) and (max-width: 640px) {
```

```
.bloc {
```

```
    display: none;
```

```
}
```

```
}
```

# LES ANIMATIONS

- ▶ Il s'agit d'une modification de la balise

css

```
.element {  
    animation: pulse 5s infinite;  
}  
  
@keyframes pulse {  
    0% {  
        background-color: #001F3F;  
    }  
    100% {  
        background-color: #FF4136;  
    }  
}
```

# LES TRANSITIONS

- ▶ Il s'agit d'une transition entre deux états différents

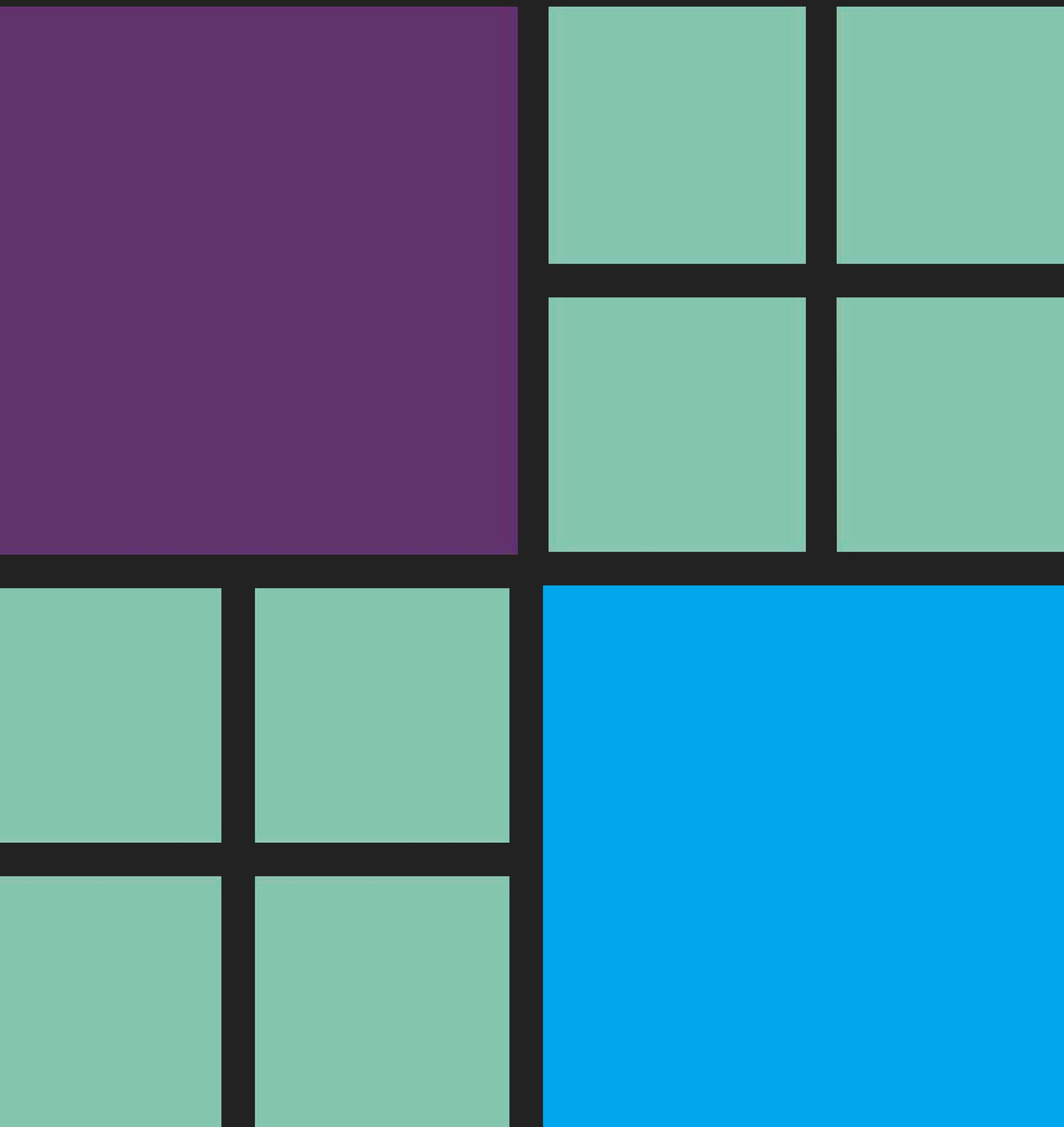
css

```
div {  
    transition: background-color 0.5s ease;  
    background-color: red;  
}  
div:hover {  
    background-color: green;  
}
```

# FLEX

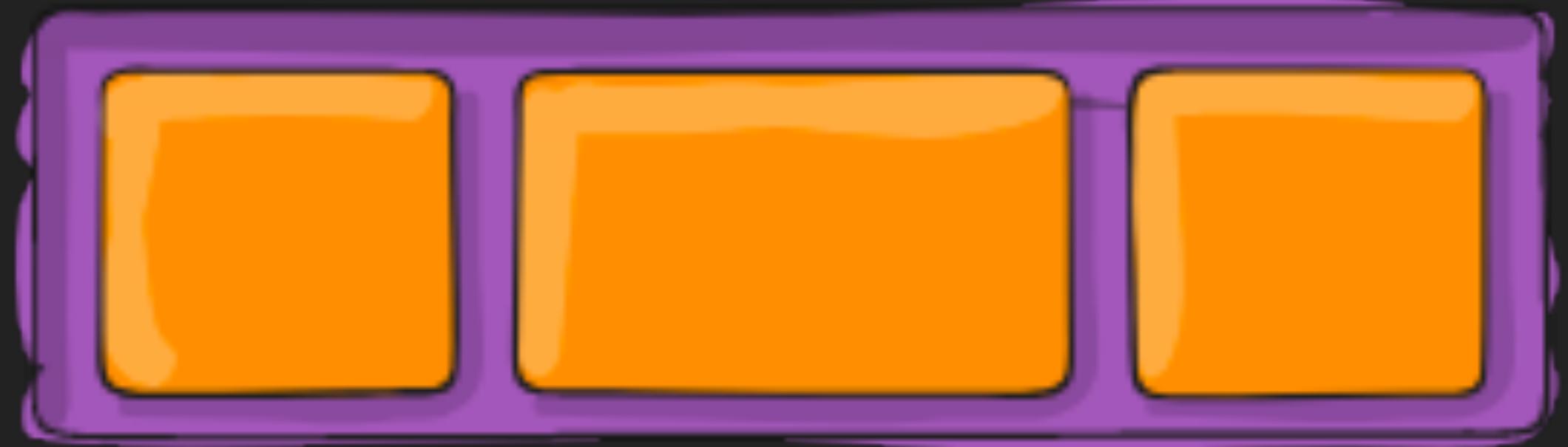
# HISTOIRE

- ▶ W3C Octobre 2017

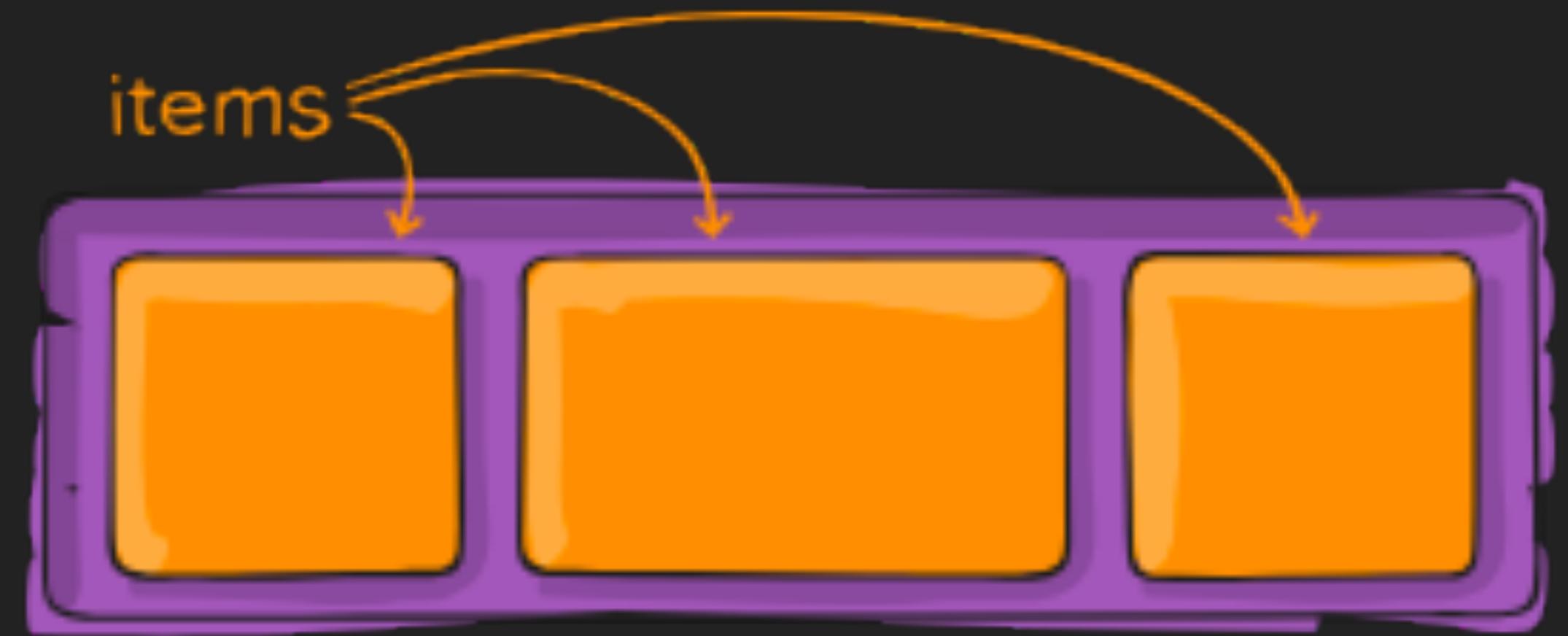


## LES BASES

Parent  
container



Enfants



CSS

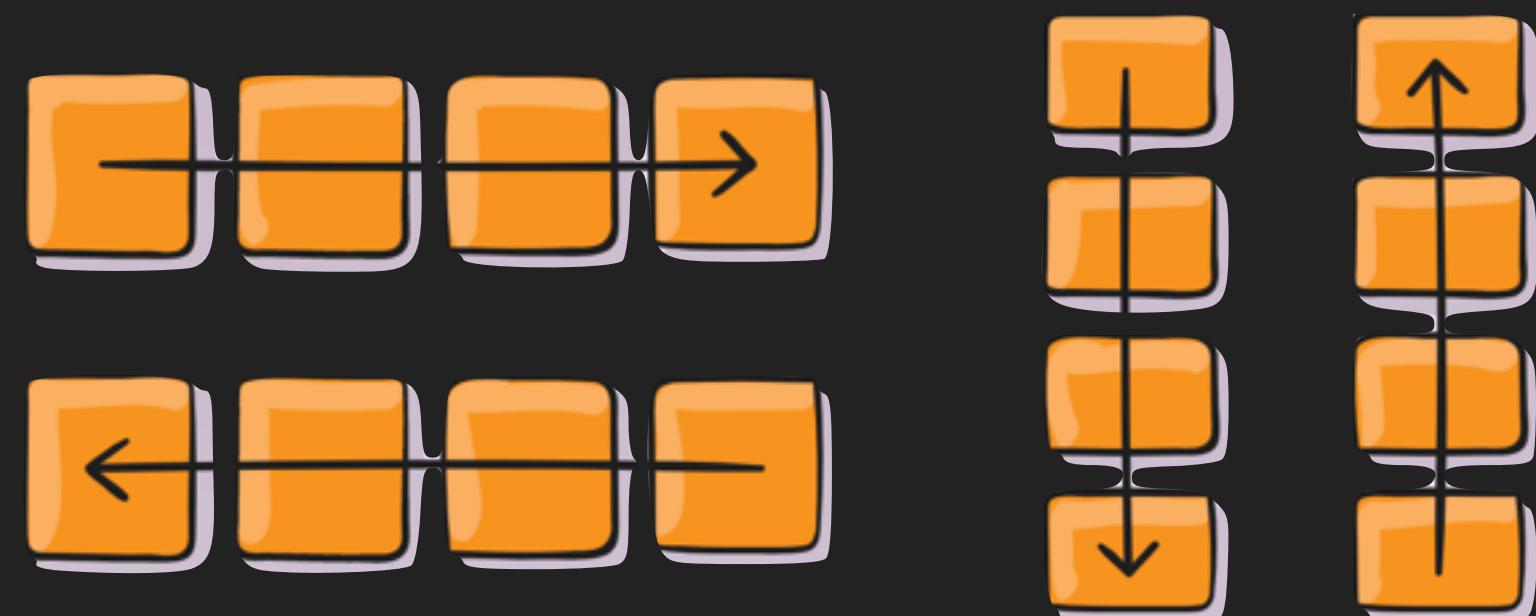
```
.container {  
  display: flex; /* or inline-flex */  
}
```

CSS

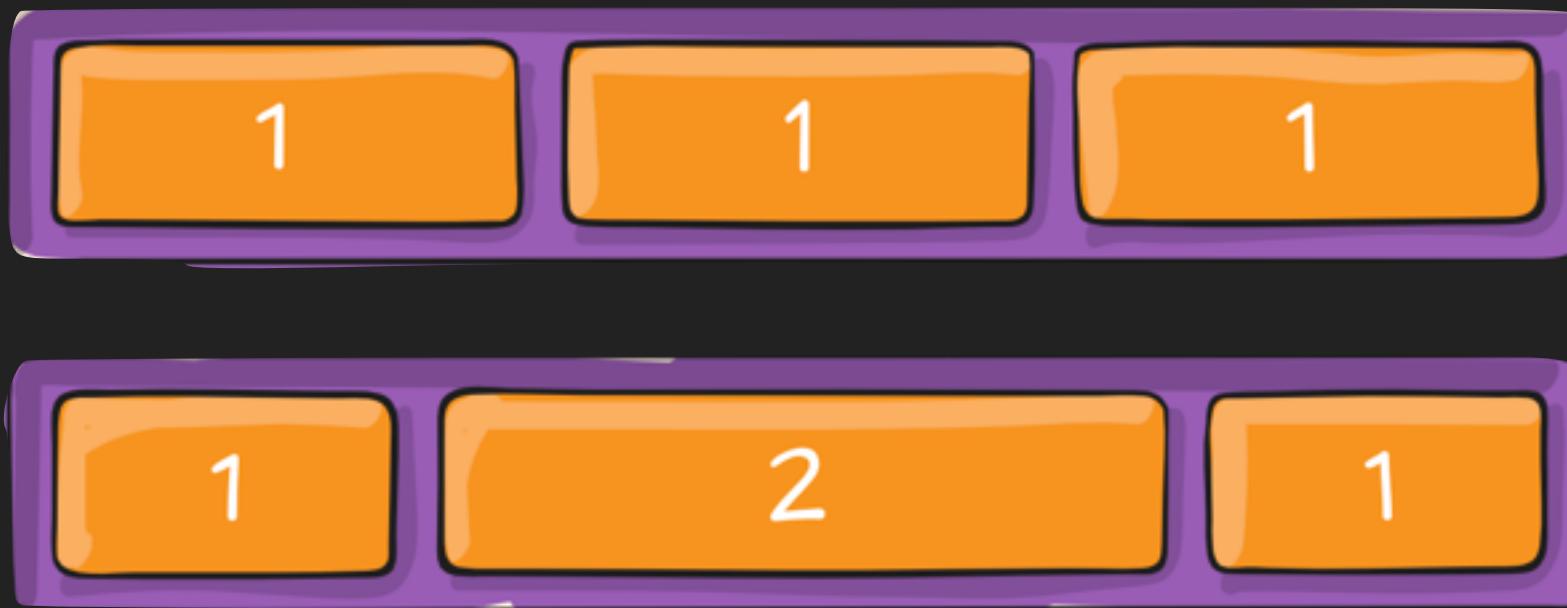
```
.item {  
  order: <integer>; /* default is 0 */  
}
```

# LES PROPRIÉTÉS 1/5

Direction



Poids

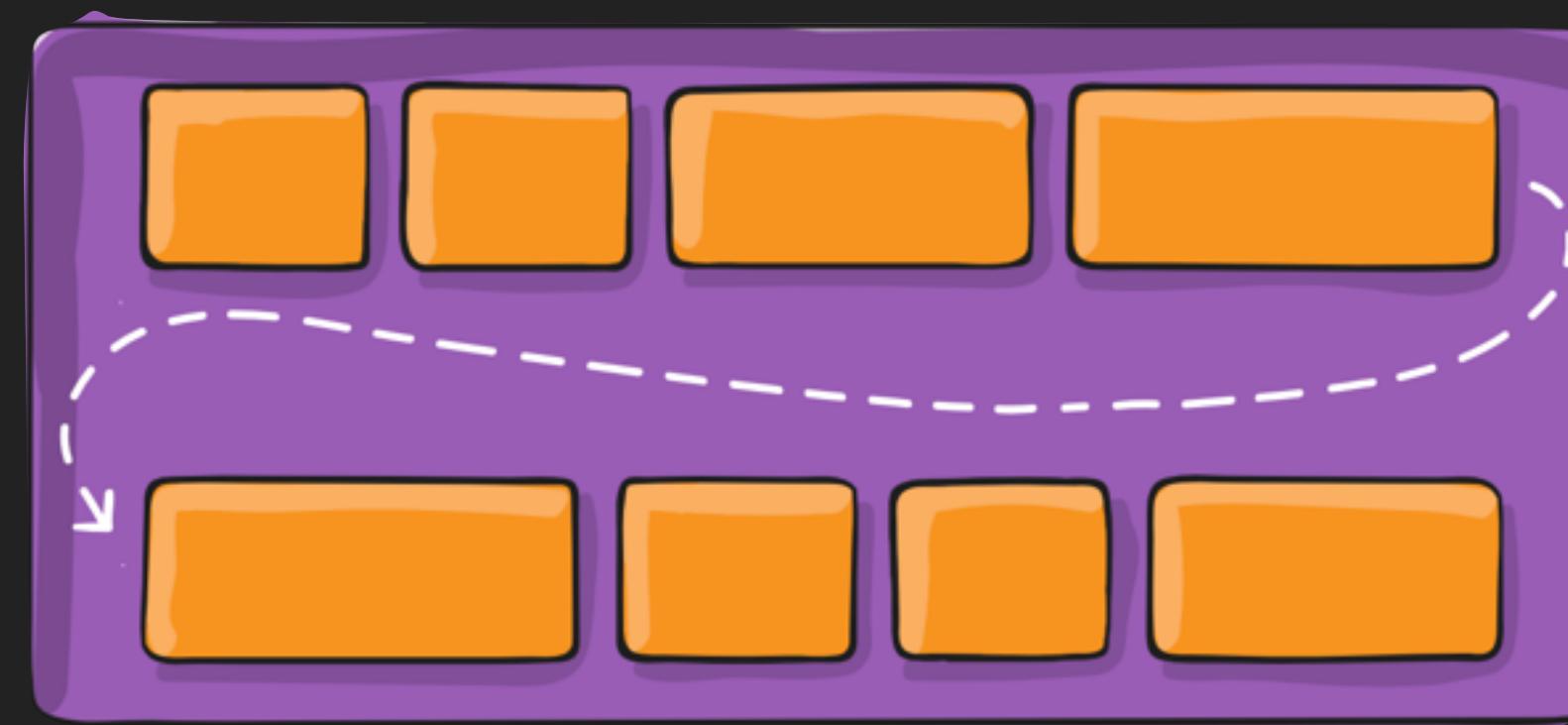


```
css
.container {
  flex-direction: row | row-reverse | column | column-reverse;
}
```

```
css
.item {
  flex-grow: <number>; /* default 0 */
}
```

## LES PROPRIÉTÉS 2/5

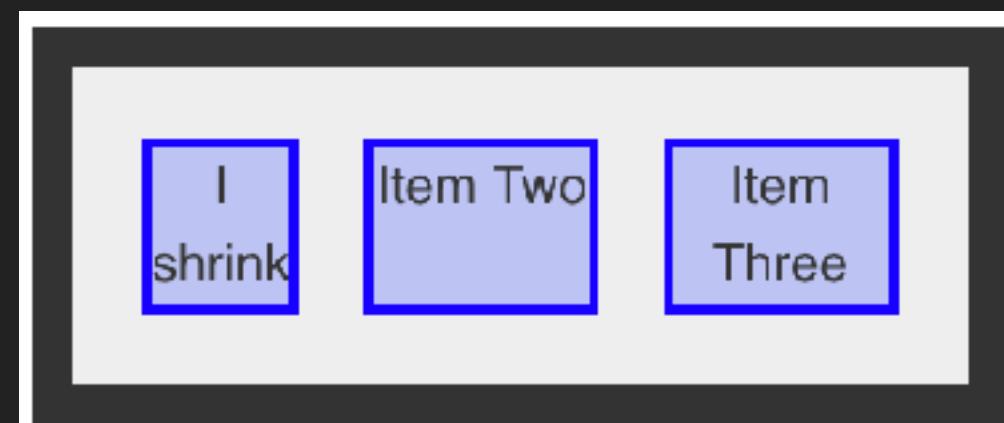
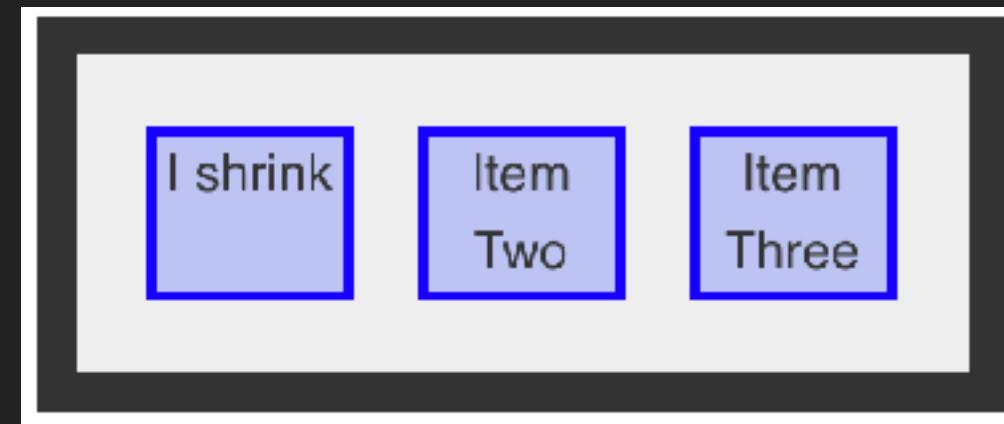
Wrap



css

```
.container{  
  flex-wrap: nowrap | wrap | wrap-reverse;  
}
```

Rétrécissement



css

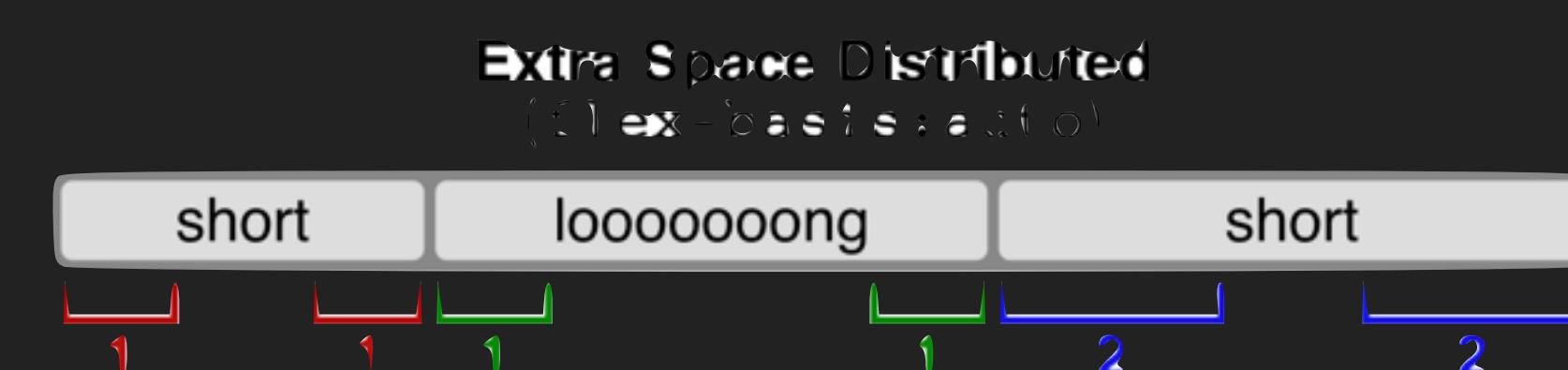
```
.item {  
  flex-shrink: <number>; /* default 1 */  
}
```

# LES PROPRIÉTÉS 3/5

Flow

Basis

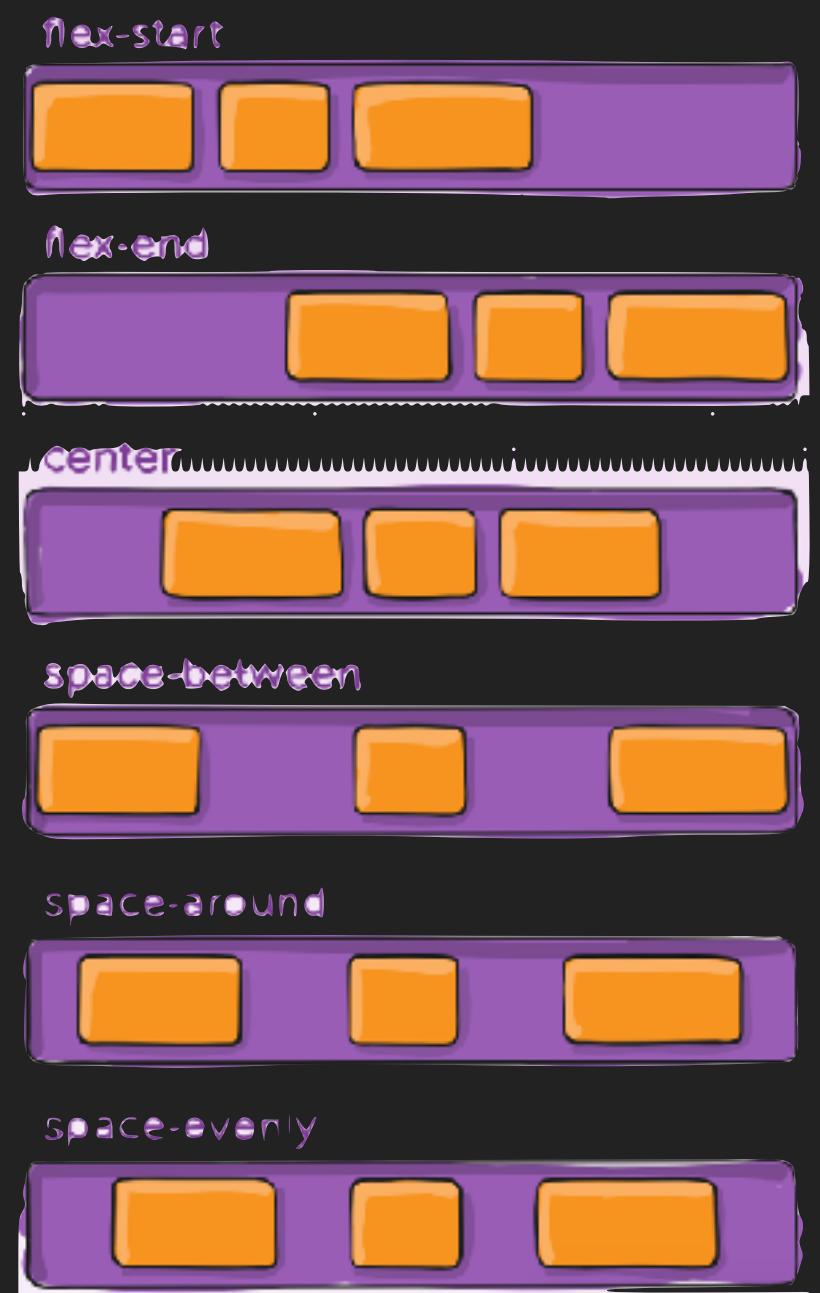
```
css
flex-flow: <'flex-direction'> || <'flex-wrap'>
```



```
css
.item {
  flex-basis: <length> | auto; /* default auto */
}
```

# LES PROPRIÉTÉS 4/5

## Justify Content



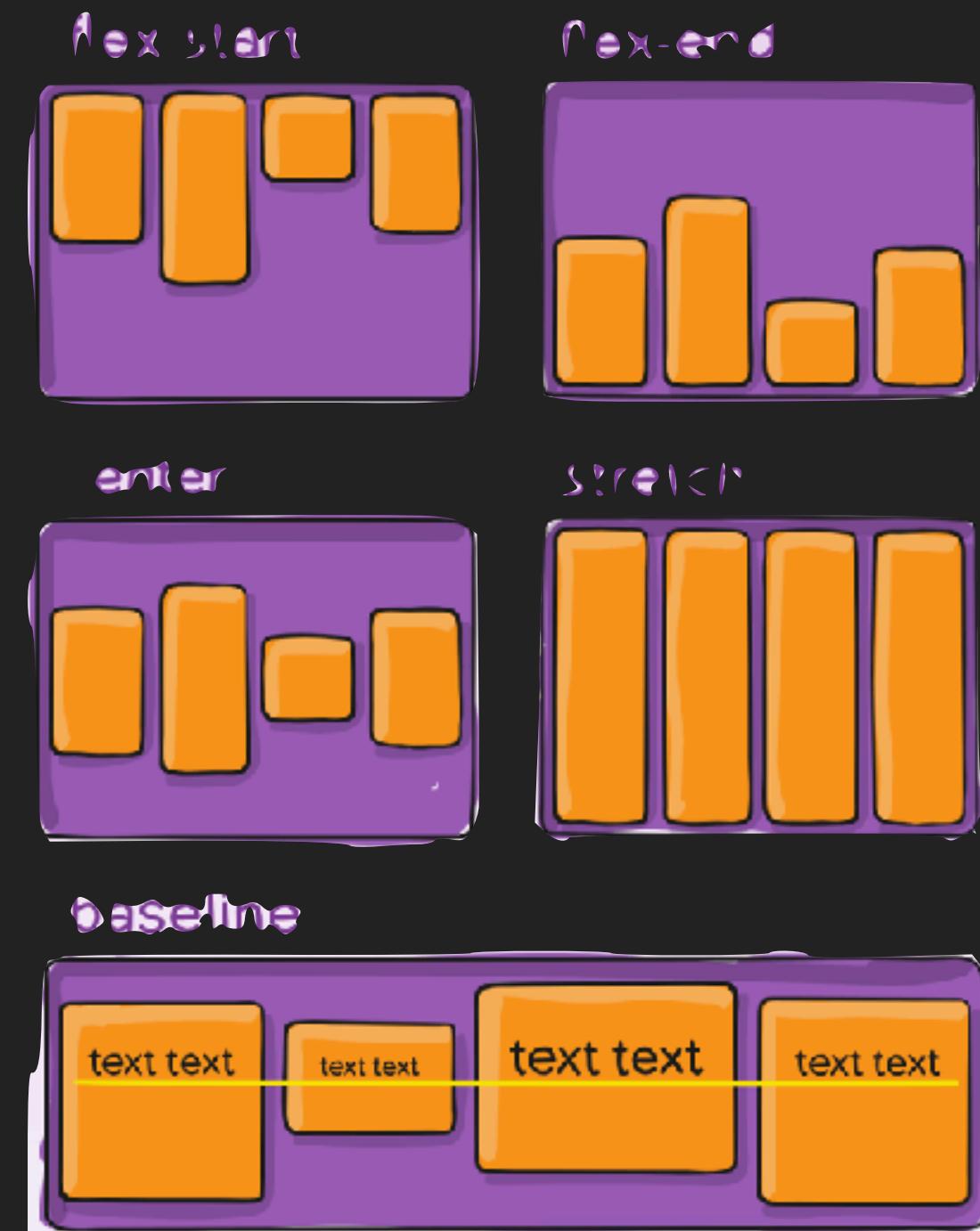
## Flex

```
css
.item {
  flex: none | [<'flex-grow'> <'flex-shrink'>? || <'flex-basis'> ]
}
```

```
css
.container {
  justify-content: flex-start | flex-end | center | space-between | space-around | sp
}
```

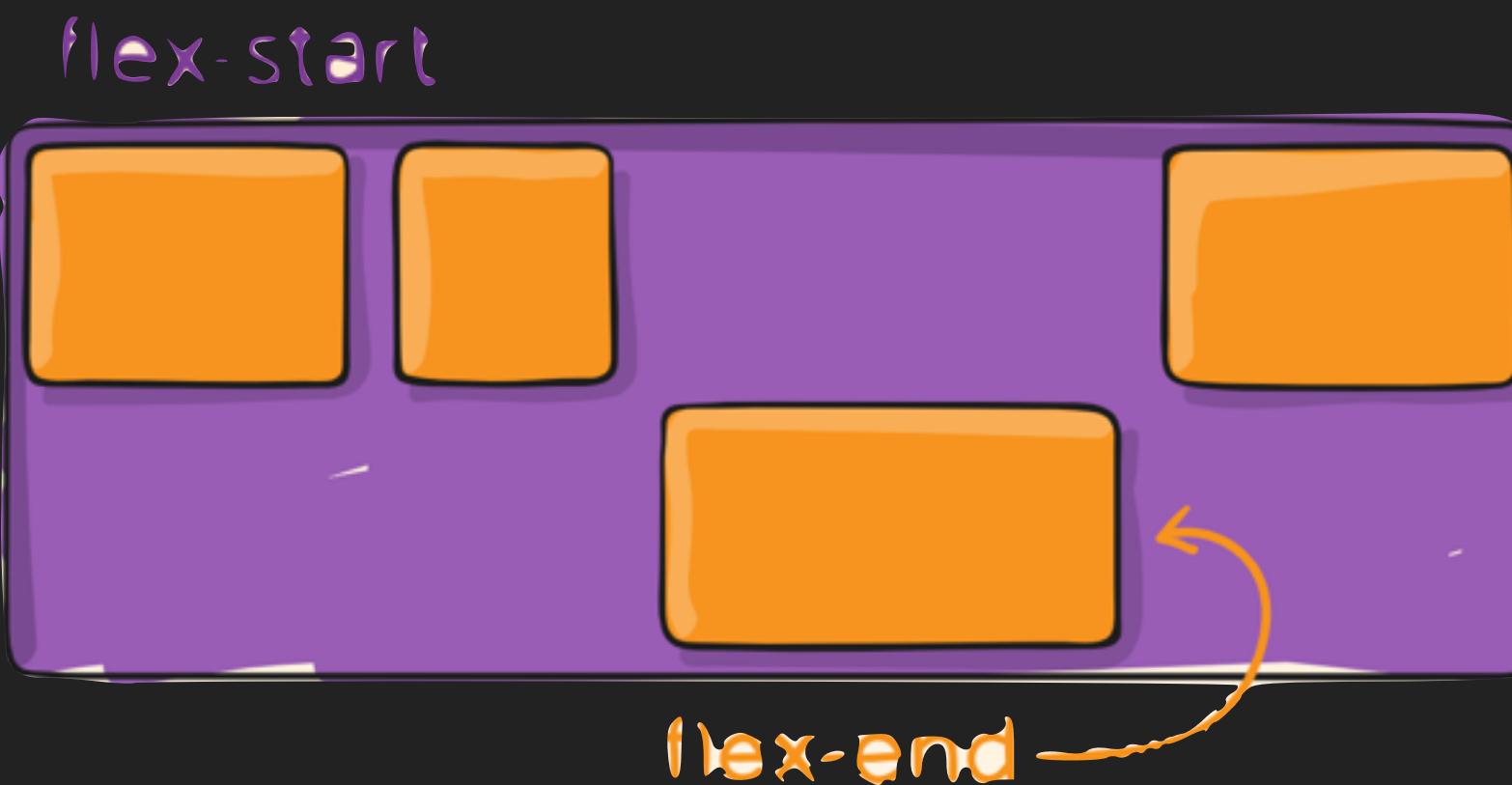
# LES PROPRIÉTÉS 5/5

## Align-Items



```
css
.container {
  align-content: flex-start | flex-end | center | space-between | space-around | stretch;
}
```

## Align-Self



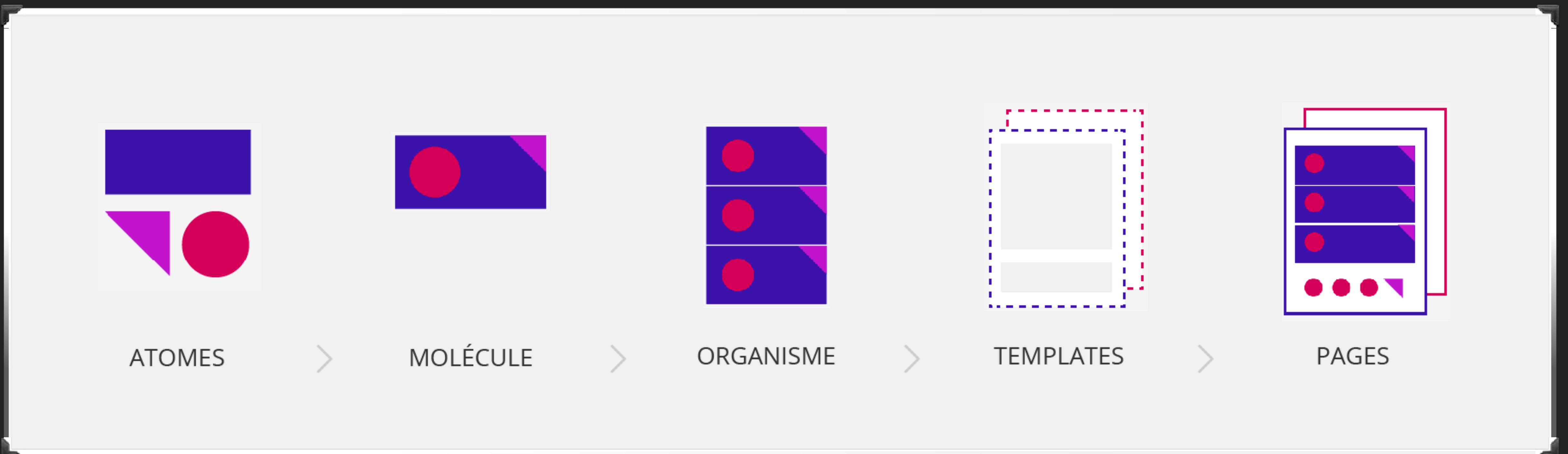
```
css
.item {
  align-self: auto | flex-start | flex-end | center | baseline | stretch;
}
```

# ATOMIC DESIGN

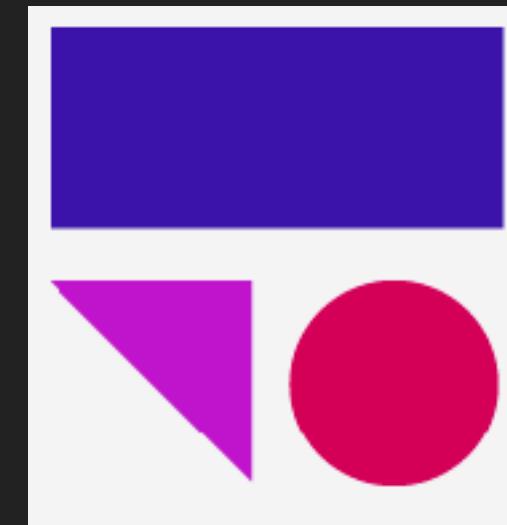
WE'RE NOT DESIGNING PAGES  
ANYMORE. WE'RE DESIGNING SYSTEMS  
OF COMPONENTS

Stephen Hay

## COMPOSITION 1/11



## COMPOSITION 2/11

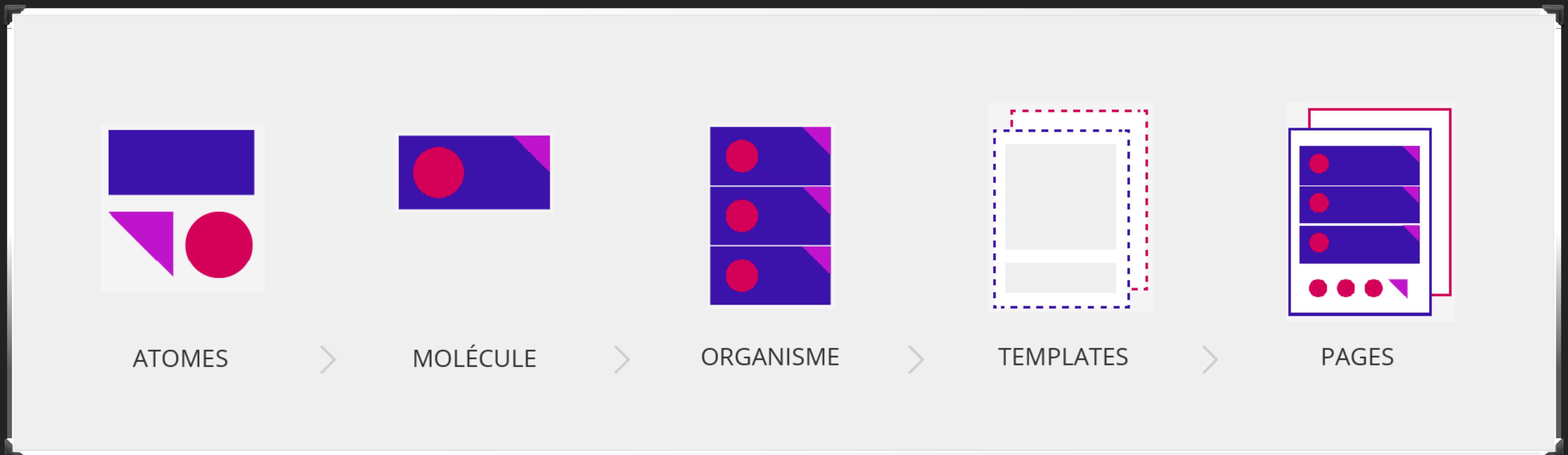


L'atome

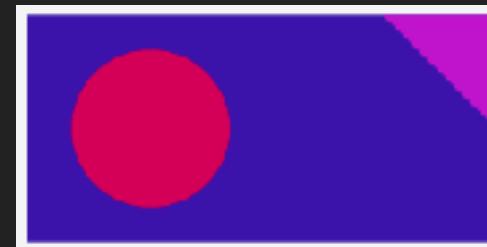
C'est un élément qui, seul, n'a pas de but fonctionnel. Il est « irréductible », ne peut pas être divisé et il compose la base de tout élément graphique de l'interface.

*Ex : un logo, une couleur, un style typographique, un bloc image, une icône, un champ de saisie...*

## COMPOSITION 3/11



## COMPOSITION 4/11



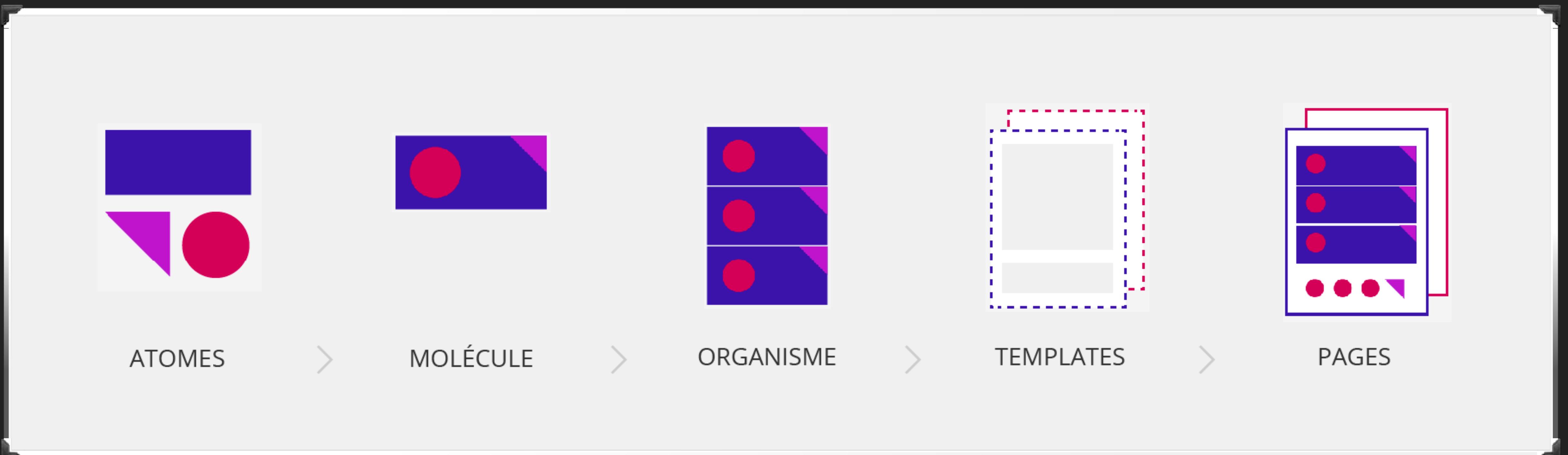
### La molécule

Ce sont des collections d'atomes qui forment des composants d'interface simples.

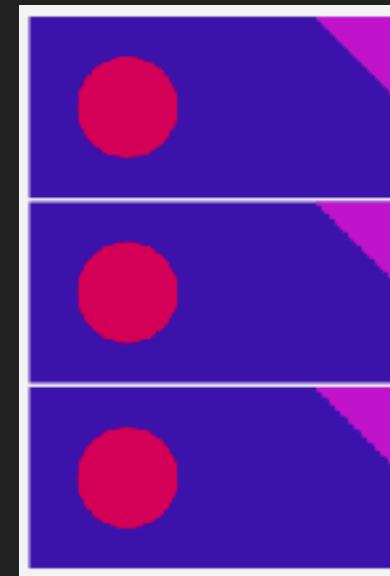
Les molécules doivent être pensées « responsive ». Il faut définir si elles sont fixes ou fluides, et sur quelles tailles de device elles apparaîtront ou non.

*Exemple : label + champ de saisie + picto loupe = champ de recherche.*

## COMPOSITION 5/11



## COMPOSITION 6/11

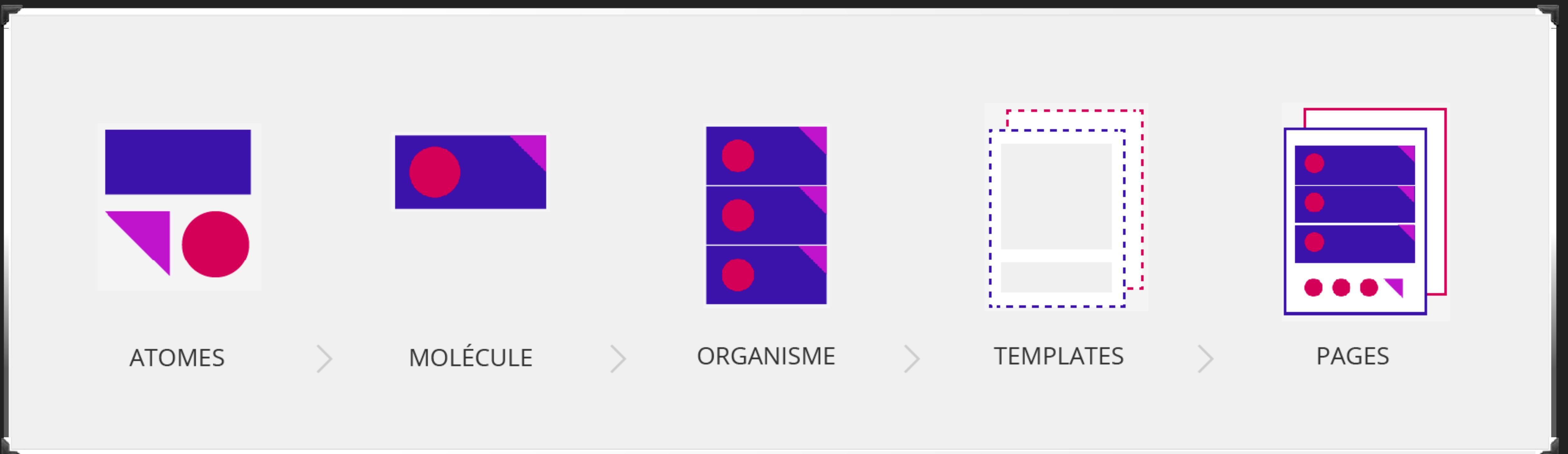


### L'organisme

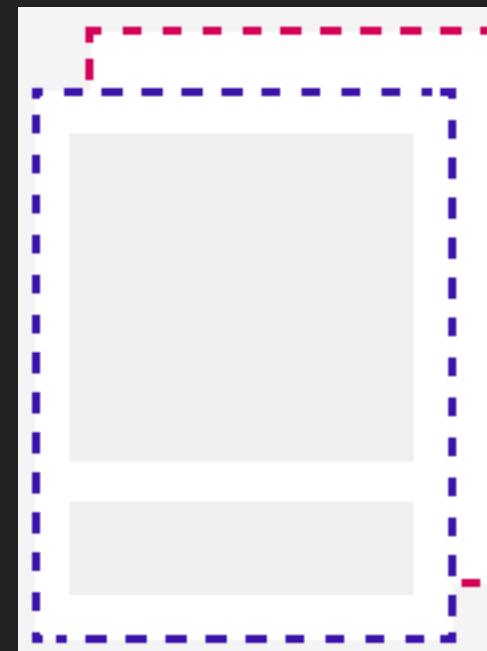
Ce sont des combinaisons plus complexes de différentes molécules ou de molécules + atomes qui forment une partie de l'interface finale.

*Ex : champ de recherche + nav + logo = header*

## COMPOSITION 7/11



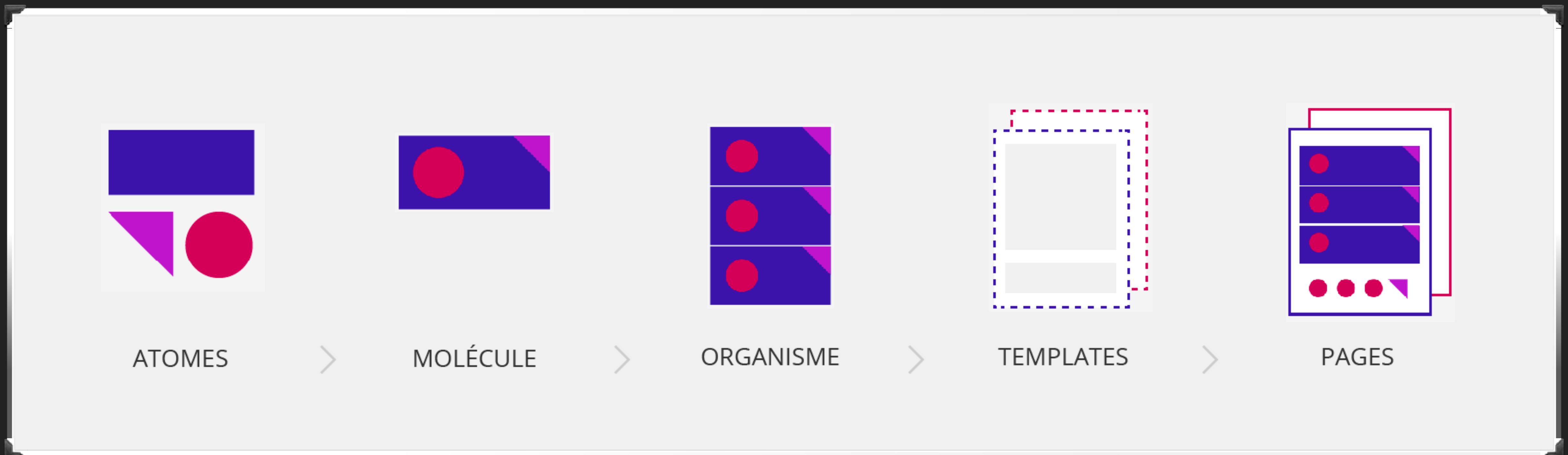
## COMPOSITION 8/11



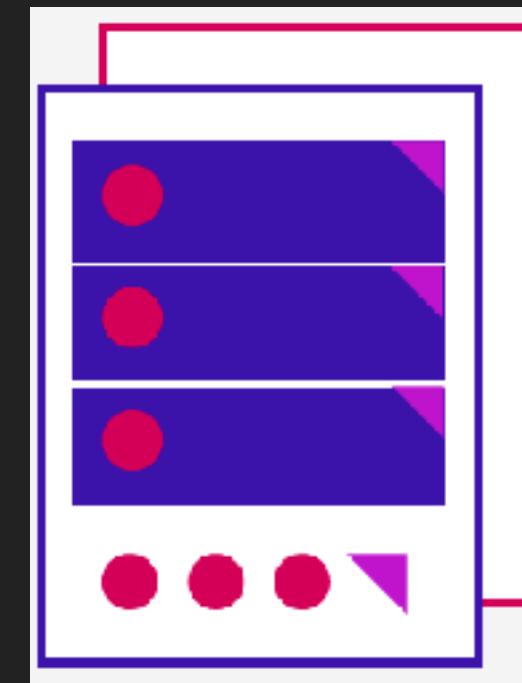
### Le template

Dans l'Atomic Design vu par Brad, les templates sont déjà développés en code. Ils peuvent être dépourvus de contenus réels (par exemple, on mettra du *lorem ipsum* à la place des textes et des placeholders pour les images ou icônes). Ils sont là pour vérifier l'organisation et la hiérarchie des divers organismes créés et de tester leurs comportements "responsive".

## COMPOSITION 9/11



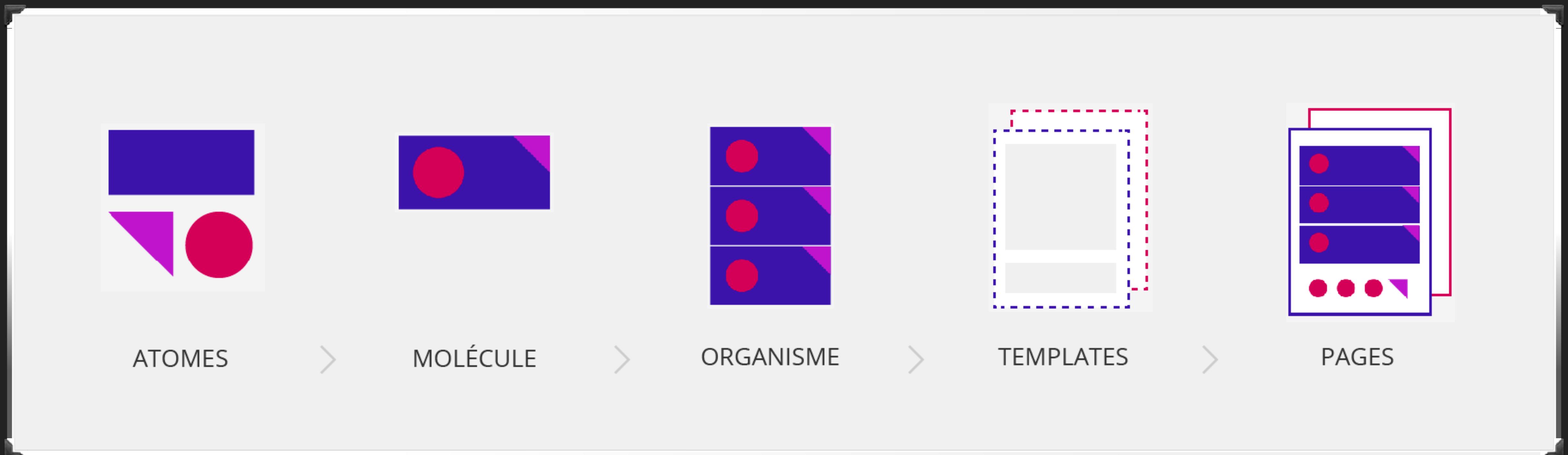
# COMPOSITION 10/11



La page

Ce sont des templates qui ont évolués vers ce que sera l'écran dans sa version finale. Tous les placeholders ont été remplacés par les vrais contenus (textes, images, couleurs, pictos, organismes et molécules finalisées...).

## COMPOSITION 11/11



# DARK MODE

## DÉCLARATION

- ▶ Notre site supporte les deux schémas de couleur

css

```
:root {  
  color-scheme: light dark; /* both supported */  
}
```

# UTILISATION DANS LE CSS

- ▶ Utilisation de variable
  - ▶ `text-color` et `bkg-color` (*L 2 et 3*)
  - ▶ Ces variables seront changées en fonction du schéma actuel
- ▶ Utilisation du media query `prefers-color-scheme` (*L 6*)
  - ▶ On surcharge les variables dans le cas où la query est satisfaite (*L 9 et 10*)
- ▶ Application des variables où il se doit (*L 16 et 21*)

```
1  body {  
2      --text-color: #222;  
3      --bkg-color: #fff;  
4  }  
5  
6  @media (prefers-color-scheme: dark) {  
7      /* defaults to dark theme */  
8      body {  
9          --text-color: #eee;  
10         --bkg-color: #121212;  
11     }  
12 }  
13  
14  
15 body {  
16     background: var(--bkg-color);  
17 }  
18  
19 h1,  
20 p {  
21     color: var(--text-color);  
22 }
```

# UTILISATION POUR LES IMAGES - CSS

css

```
/* Apply the filter directly on the body tag */
body.dark-theme img {
    filter: brightness(.8) contrast(1.2);
}

/* Or apply it via media query */
@media (prefers-color-scheme: dark) {
    img {
        filter: brightness(.8) contrast(1.2);
    }
}
```

## UTILISATION POUR LES IMAGES - PICTURE/SOURCE

HTML

```
<picture>
  <!-- Use this image if the user's OS setting is light or unset -->
  <source srcset="photo-light.png" media="(prefers-color-scheme: light) or (prefers-color-scheme: no-preference)">
  <!-- Use this image if the user's OS setting is dark -->
  <source srcset="photo-dark.png" media="(prefers-color-scheme: dark)">
</picture>
```

## EXEMPLES

- ▶ Source : <https://github.com/j-catania/apple-darkmode-example>
  - ▶ Rendu : <https://j-catania.github.io/apple-darkmode-example/>
- ▶ YouTube -> utilisation d'un attribut HTML dans le *body* et changement des variables dans un sélecteur CSS *body[dark]*
- ▶ DuckDuckGo
- ▶ Slack
- ▶ Twitter -> déclaration de plusieurs thèmes
- ▶ ...

# JAVASCRIPT OBJECT NOTATION

- ▶ Array : [...]
  - ▶ [1,2,4]
- ▶ Objet : {...}
  - ▶ {"nom":"catania","age":42}
- ▶ Validator : **JSONLint** - *The JSON Validator*

**Git**

# COMMAND LINES

- ❖ `git clone <url>`
  - ❖ Récupération du code source depuis un repository distant
- ❖ `git add <files>`
  - ❖ Ajoute les changements à l'*index*
- ❖ `git commit -m "<message>"`
  - ❖ Valide les changements de l'*index*
- ❖ `git push <remote> <branch>`
  - ❖ *Envoi les commits au repository distant*



# LE FRAMEWORK

ONE CODEBASE.  
ANY PLATFORM.

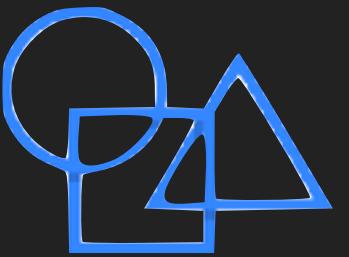
Ionic TEAM

## HISTORIQUE DE VERSIONS

- ▶ V1.0.0 - 12/05/2015
- ▶ V2.0.0 - 22/01/2017
- ▶ V3.7.1 - 29/09/2017
- ▶ V4.11.1 - 14/10/2019
- ▶ V5.6.14 - 18/08/2021
- ▶ V6.3.0 - 07/10/2022

# ARCHITECTURE

- ▶ Ionic UI Component



- ▶ Angular/VueJS/React & Ionic API



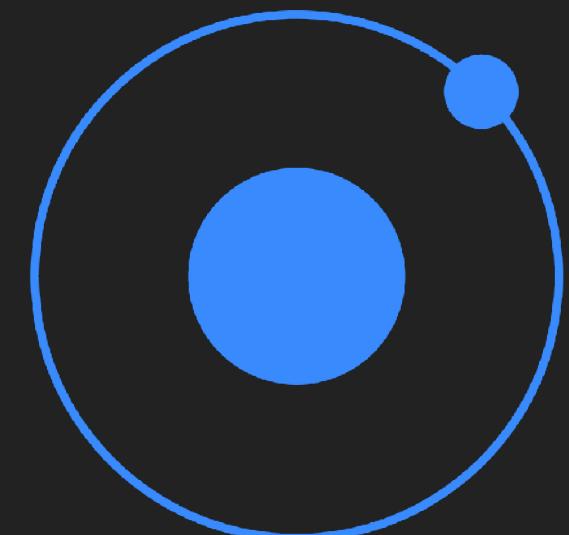
- ▶ Ionic Native



- ▶ Capacitor



# ARCHITECTURE



# MARKET

MARKET

---

## STARTERS, PLUGINS & THEMES

[HTTPS://MARKET.IONICFRAMEWORK.COM/](https://market.ionicframework.com/)

# STARTERS

Starters

Featured   Newest   View all



**Ionic Starter Messenger - Firebase Integrate**  
Building up your social media app quickly and beautifully with abundant templates



**TypeScript Starter**  
A starter project that brings TypeScript's type safety, code insight, and rich IDE support to Angular and Ionic!



**Ionic V5: Ionic Message Chat, Video Call, Live Stream** ios, android, desktop app, web  
Window Macos Ubuntu Ios Android web



**Animated Grid**  
Animated Grid Starter App  
★★★★★ (2)

# PLUGINS

Plugins

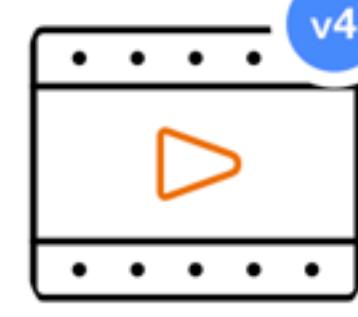
Featured   Newest   View all



Lumen Sidemenu

Easily create your own unique sidemenu!

★★★★★ (3)



Onymos Media

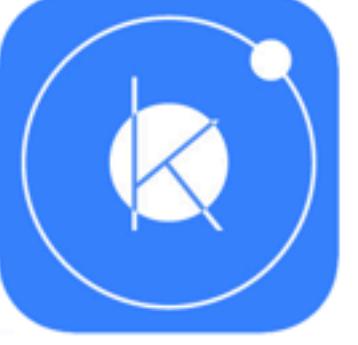
Add Photo/Video feature to your Ionic app in minutes



IonPhaser WebComponent

A web component to integrate Phaser Framework with Angular, React, Vue, etc

★★★★★ (1)



Ionic Digital Keyboard

A digital/numeric keyboard component for your Ionic apps

★★★★★ (4)

## THEMES

Themes

Featured      Newest      View all



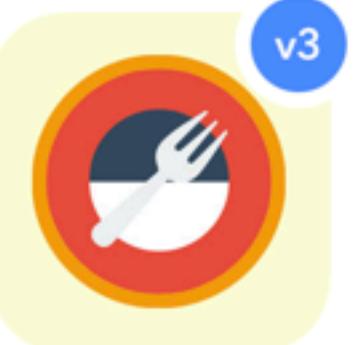
Tinder React 5  
Ionic React 5



Cross  
A social community with videos sharing theme for ionic app



Grocery World  
An advanced sass based ionic template for online shop.



Foodica  
Mobile/Tablet template for ionicframework

# APPFLOW

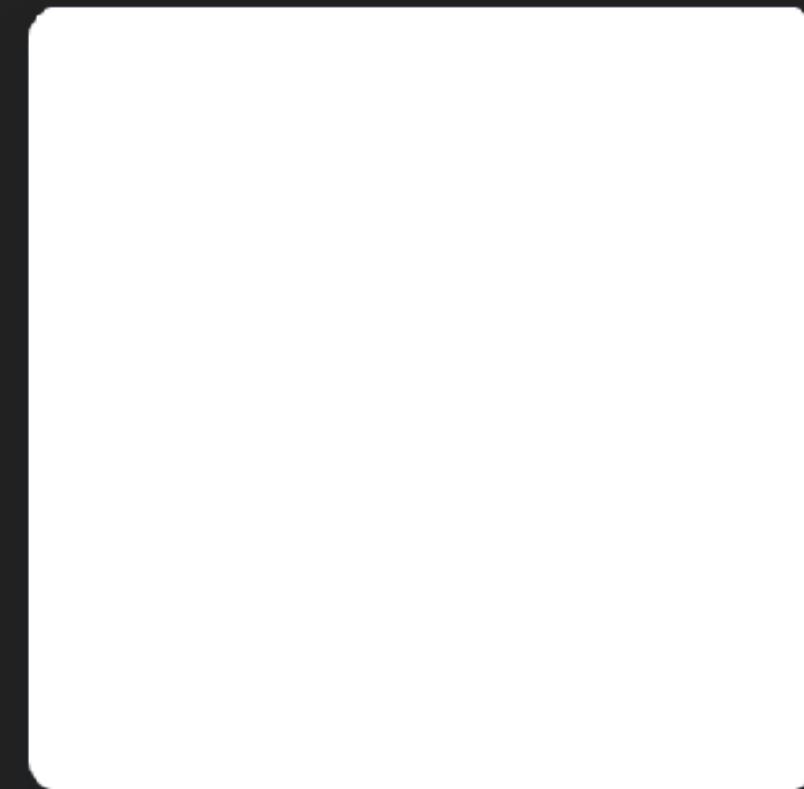
## LES SERVICES

- ▶ Repository git
- ▶ Déploiement auto
- ▶ Aperçu instantané
- ▶ Partage aux collaborateurs
- ▶ Tracker d'erreur
- ▶ Build final natif

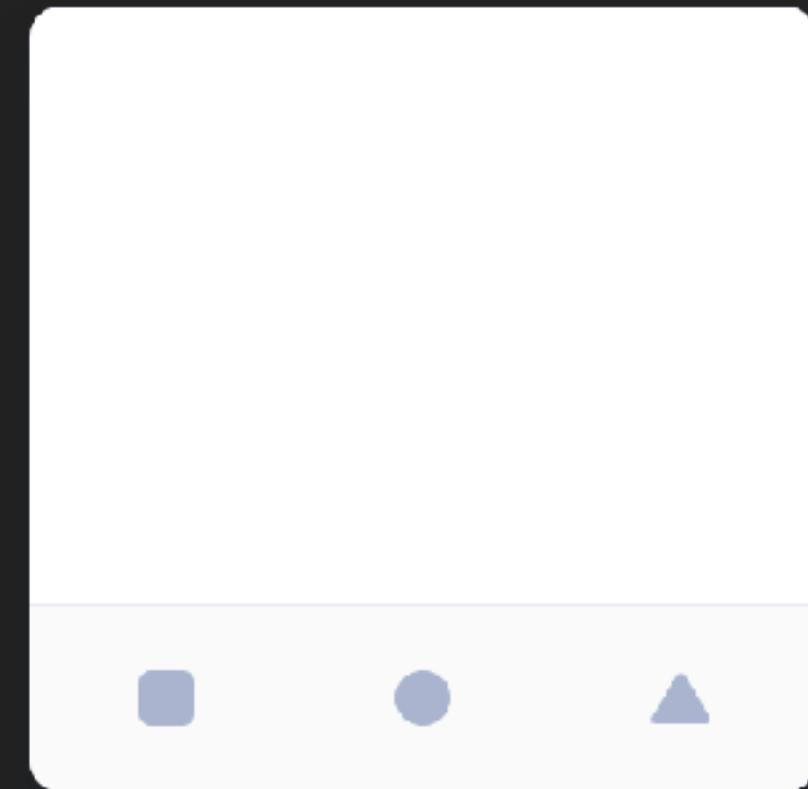
# CRÉATION DU WORKSPACE

# INSTALLATION

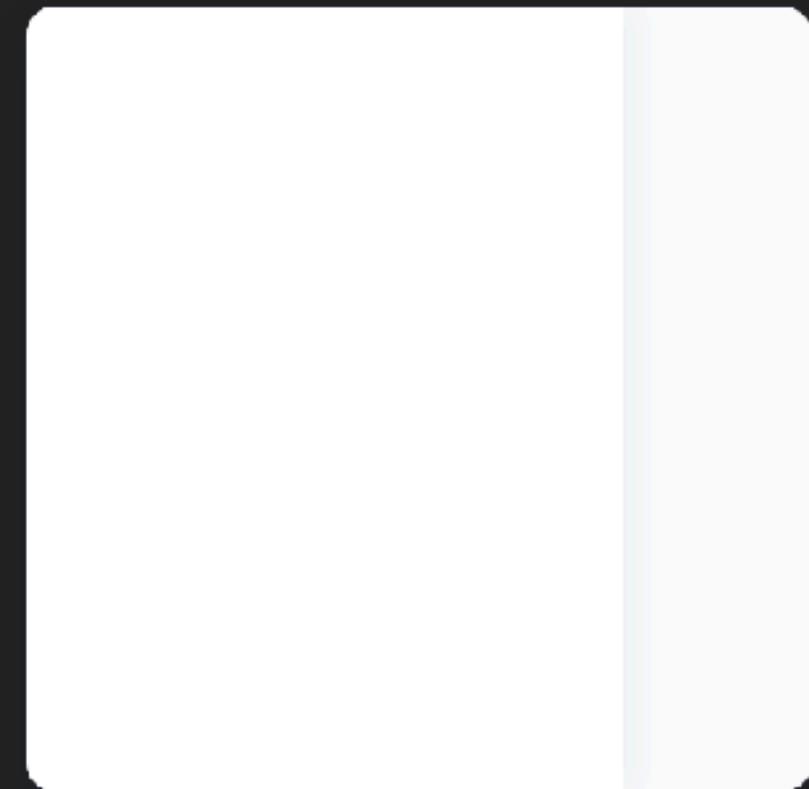
```
$ npm install -g @ionic/cli
```



BLANK



TABS



SIDE MENU

```
$ cd myApp  
$ ionic serve
```

# INITIALISATION D'UNE APP - STARTERS

```
→ ~ ionic start --list
```

## Starters for @ionic/vue (--type=vue)

### name | description

tabs	A starting project with a simple tabbed interface
sidemenu	A starting project with a side menu with navigation in the content area
blank	A blank starter project
list	A starting project with a list

## Starters for @ionic/angular (--type=angular)

### name | description

tabs	A starting project with a simple tabbed interface
sidemenu	A starting project with a side menu with navigation in the content area
blank	A blank starter project
list	A starting project with a list
my-first-app	An example application that builds a camera with gallery
conference	A kitchen-sink application that shows off all Ionic has to offer

## Starters for @ionic/react (--type=react)

### name | description

blank	A blank starter project
list	A starting project with a list
my-first-app	An example application that builds a camera with gallery
sidemenu	A starting project with a side menu with navigation in the content area
tabs	A starting project with a simple tabbed interface
conference	A kitchen-sink application that shows off all Ionic has to offer

## Starters for Ionic 2/3 (--type=ionic-angular)

### name | description

tabs	A starting project with a simple tabbed interface
sidemenu	A starting project with a side menu with navigation in the content area
blank	A blank starter project
super	A starting project complete with pre-built pages, providers and best practices for Ionic development.
tutorial	A tutorial based project that goes along with the Ionic documentation
aws	AWS Mobile Hub Starter

## Starters for Ionic 1 (--type=ionic1)

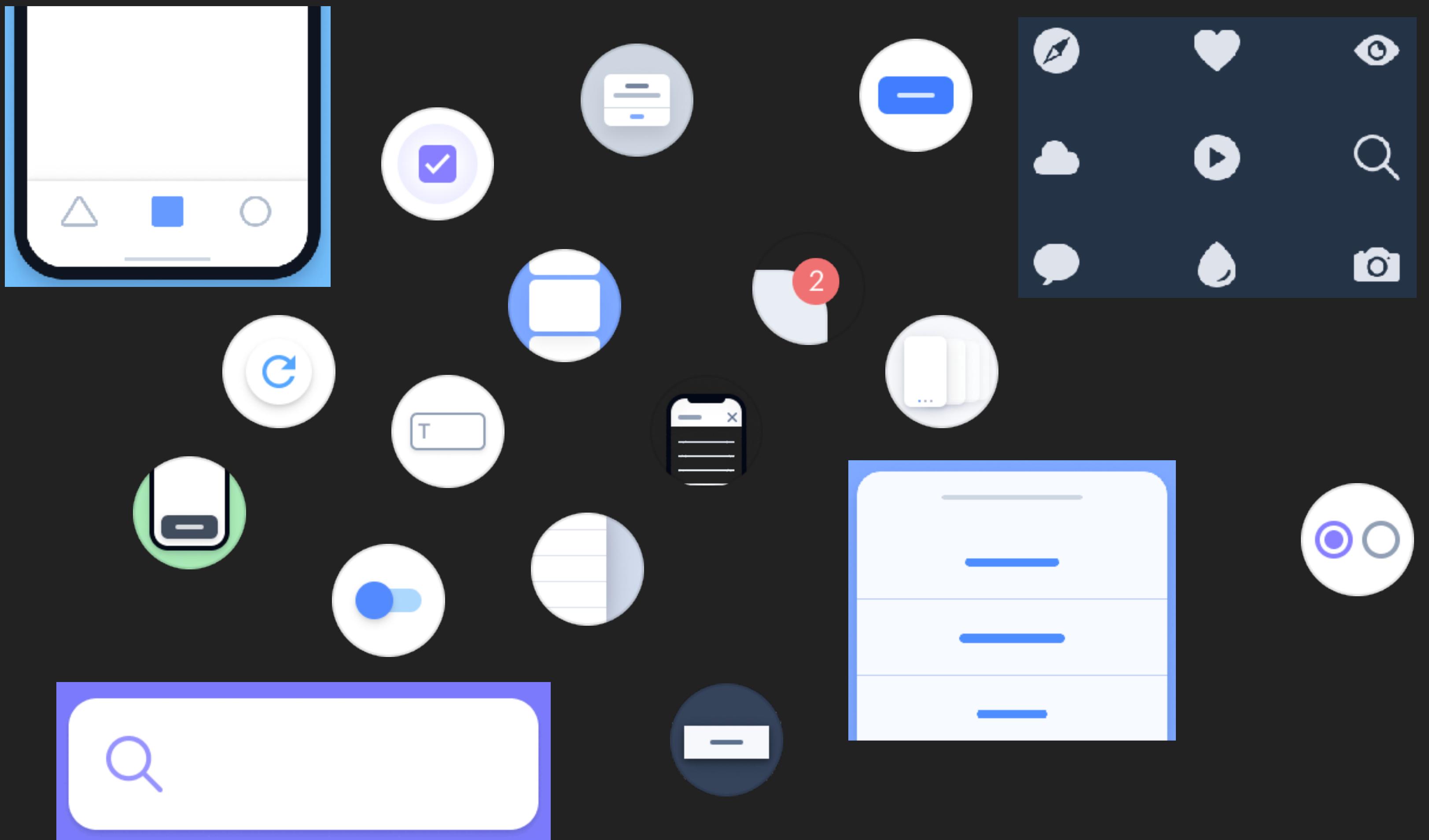
### name | description

tabs	A starting project for Ionic using a simple tabbed interface
sidemenu	A starting project for Ionic using a side menu with navigation in the content area
blank	A blank starter project for Ionic
maps	An Ionic starter project using Google Maps and a side menu

# UI COMPONENTS

# COMPOSANTS GRAPHIQUES

- ▶ Alert
- ▶ Button
- ▶ Card
- ▶ Gesture
- ▶ Grid
- ▶ Icons
- ▶ Popover
- ▶ Slides
- ▶ Toast



# API NATIVE

# INSTALLATION D'UN PLUGIN

## ▶ Promises and Observables

```
// Install Ionic Native TypeScript wrapper
$ npm install @ionic-native/camera

// Install Cordova plugin
$ npm install cordova-plugin-camera

// Update native platform project(s) to include newly added plugin
$ ionic cap sync
```

# UTILISATION D'UN PLUGIN SOUS ANGULAR

Ajout du Provider

```
// camera.service.ts
import { Injectable } from "@angular/core";
import { Camera, CameraOptions } from "@ionic-native/camera/ngx";

@Injectable({
  providedIn: "root",
})
export class PhotoService {
  constructor(private camera: Camera) {}

  ...
  takePicture() {
    const options: CameraOptions = {
      quality: 100,
      destinationType: this.camera.DestinationType.DATA_URL,
      encodingType: this.camera.EncodingType.JPEG,
      mediaType: this.camera.MediaType.PICTURE,
    };

    this.camera.getPicture(options).then(
      (imageData) => {
        // Do something with the new photo
      },
      (err) => {
        // Handle error
        console.log("Camera issue: " + err);
      }
    );
  }
}
```

TYPESCRIPT

SCRIPT

Injection et utilisation du service

# UTILISATION D'UN PLUGIN SOUS REACT

TYPESCRIPT

```
import { BarcodeScanner } from "@ionic-native/barcode-scanner";

const Tab1: React.FC = () => {
  const openScanner = async () => {
    const data = await BarcodeScanner.scan();
    console.log(`Barcode data: ${data.text}`);
  };
  return (
    <IonPage>
      <IonHeader>
        <IonToolbar>
          <IonTitle>Tab 1</IonTitle>
        </IonToolbar>
      </IonHeader>
      <IonContent>
        <IonButton onClick={openScanner}>Scan barcode</IonButton>
      </IonContent>
    </IonPage>
  );
};
```

# API NATIVE

- ▶ 3D Touch
- ▶ AdMod
- ▶ Android Fingerprint Auth
- ▶ App Availability
- ▶ App Version
- ▶ Autostart
- ▶ Badge
- ▶ Base64 to Galery
- ▶ Bluetooth
- ▶ Camera
- ▶ Date Picker
- ▶ Device
- ▶ Fingerprint AIO
- ▶ Firebase Cloud Messaging
- ▶ Geolocation
- ▶ Gyroscope
- ▶ Health / Health Kit
- ▶ Local Notifications
- ▶ Market
- ▶ Touch ID

# REAL DEVICE

- ▶ Pré-requis
  - ▶ iOS
    - ▶ Xcode
    - ▶ macOS
  - ▶ Android
    - ▶ Android SDK
- ▶ Ajout de la plateforme dans le projet
  - ▶ *ionic capacitor add ios|android*
- ▶ Build du projet sur le mobile
  - ▶ *ionic capacitor build ios|android*
- ▶ Run du projet sur le Device
  - ▶ *ionic capacitor run ios|android --livereload*